

Appl. No.: 10/735,003
Amdt. dated August 29, 2008
Reply to Office action July 29, 2008

Amendments to the Claims:

This listing of the claims will replace all prior versions, and listings, of the claims in the application.

Listing of Claims:

Please amend the claims as follows without prejudice. No new matter has been added by way of these amendments.

We claim:

Claim 1 (Canceled)
Claim 2 (Canceled)
Claim 3 (Canceled)
Claim 4 (Canceled)
Claim 5 (Canceled)
Claim 6 (Canceled)
Claim 7 (Canceled)
Claim 8 (Canceled)
Claim 9 (Canceled)
Claim 10 (Canceled)
Claim 11 (Canceled)
Claim 12 (Canceled)
Claim 13 (Canceled)
Claim 14 (Canceled)
Claim 15 (Canceled)
Claim 16 (Canceled)
Claim 17 (Canceled)

Claim 18 (Previously presented) A method external from a native operating system of a computer system for generating a file identifier for use in controlling access to file system resources in a computer system comprising the steps of:

generating a data structure having a pointer to an index related to a physical location of a file resource in the file system and a pointer to a directory containing a requested file resource;

obtaining a unique physical attribute of the file system object, by retrieving a serial number for a file location where a requested file system resource resides the file system location is retrieved from an inode or vnode index or from a serial number generated using a programming interface;

obtaining the name of the file system object, by opening the directory identified in the data structure, for each entry in the directory, reading the serial number for the file location, comparing said vnode or inode number to said serial number for the file location number and retrieving the file name of the resource out of the directory; and

constructing a file identifier for that file system object from said obtained unique physical attribute and said file system object name, by coupling the unique physical attribute to the file system object name to produce the file identifier for a particular file system object.

Claim 19 (Canceled)

Claim 20 (Canceled)

Claim 21 (Canceled)

Claim 22 (Canceled)

Claim 23 (Original) The method as described in claim 22 wherein said file identifier construction step comprises placing the index number at the beginning of the of bytes that will be the file identifier and appending the file name to the file of bytes.

Claim 24 (Original) The method as described in claim 22 further comprising after said comparing step, the steps of: retrieving the next entry in the directory when the said comparison is not equal; determining if this entry is the last entry; and proceeding to read said entry is not the last entry.

Appl. No.: 10/735,003
Amdt. dated August 29, 2008
Reply to Office action July 29, 2008

Claim 25 (Previously presented) The method as described in claim 24 further comprising the step of returning no file identifier when no directory entry file serial number equals the index file space number

Claim 26 (Canceled)

Claim 27 (Canceled)

Claim 28 (Canceled)

Claim 29 (Canceled)

Claim 30 (Canceled)

Claim 31 (Canceled)

Claim 32 (Canceled)

Claim 33 (New) A method external from a native operating system of a computer system for generating a file identifier for use in controlling access to file system resources in the computer system, the external file generating method comprising the steps:

obtaining an underlying object data pointer for a file system object;

obtaining an object pointer of for the parent directory of the file system object;

obtaining a physical file location number for the system object using the obtained underlying object data pointer for the system file object and the obtained object pointer of for the parent directory of the file system object;

obtaining the file name for the system file object by using the data pointer for the parent directory by using the data pointer for the parent directory to open the directory;

obtaining a specific file serial number for the system file object using the physical file;

reading information contained in the directory to get a directory entry's serial number or node index;

determining whether there is a match between a serial number of an entry in the parent directory and the obtained serial number of the specific file system object;

when the determination is that the serial number of a directory entry is equal to the serial number of the specific file system object, retrieving the name of the entry in the directory and the length of that name;

constructing a file identifier for a file system resource of the file system object by placing the file serial number at the beginning of the file of bytes;

appending the length of the file name and file name to the constructed file identifier resulting in memory locations containing the serial number, the length of the file name and the file name in consecutive memory locations;

setting a file identifier length for the constructed file identifier to be equal to the length of the file name plus the number of bytes required for the serial number plus the number of bytes required to write the length of the file name; and

returning the newly constructed file identifier to the user.